IETF 103 – DTN WG

Terra - BPBis Implementation Report

Lucien Loiseau

November 8, 2018

Context and Network Assumptions

- RightMesh mission is to connect underserved areas
- One approach studied is to use the physical mobility of devices to mule data → DTN
  - High delays
  - Highly dynamic network
    - Predictable regions can serve as backbone (daily bus/ferry boat/coach/train/plane)
    - Unpredictable regions using smartphones can connect « the last mile »
- DTN Nodes are thus assumed to be mostly consumers devices → Java

Terra Features (as of Nov. 2018)

- Implementation started August 2018
- Full Java implementation with few dependencies (~14000 LOC / ~ 3 MB packaged)
- Modular extensible architecture (can load plugins at runtime)
- Drafts implemented :
  - Draft-ietf-dtn-bpbis-11
  - Draft-burleigh-dtn-stcp-00
  - Draft-ietf-dtn-bpsec-08
  - Draft-birrane-dtn-bpsec-interop-cs-03
- Code available at https://github.com/RightMesh/Terra

BPBis - Difficulties

- Administrative Record and Fragmentation is not yet implemented (lower priority)
- Some big moving part like storage and routing are not addressed (maybe in some other drafts?)
  - It is out-of-scope but bp DTN cannot be functional without those
- Afaik, no draft provides guidelines on what is the relationship between EID, routing and registration
  - Does an AA registers to an entire EID or just to the resource/service part (the « path »), or both ?
  - Does an EID identifies a service/resource or a dtn node (identifier), or both ?
- Minor Problems :
  - eid-dtn parser disjonction: either a **cbor_text_string** (dtn:*) OR **cbor_integer** (dtn:none) ← should stick to a string
  - « Retention constraint » doesn’t really have any constraint attached and is mostly informative (nowhere in the draft is retention constraint used for an IF-statement).

Implementation Decisions: AA Registration

- Endpoint ID is an URI as specified in BPBis
- Though AA may theoritically register to any EID, we limit to only a sink or resource identifier so that an EID looks like this :

```
+-------------------+-------------------------------+
| <host identifier> | <resource/service identifier> |
+-------------------+-------------------------------+
          ^                         ^
          |                         +--- identify a registration from an Application Agent
          |
          +-- identify a DTN node or a service provider
```

- Examples :
  - IPN-EID **ipn:<node>.<service>** ipn:0.0 **host=**ipn:0 **AA registration=**0
  - DTN-EID **dtn:<node>/<service>** dtn:node-1/backup/ **host=**dtn:node-1 **AA registration=**/backup/

- Draft unclear how to generically « process » EIDs ? What if two nodes with different EIDs share a same prefix ?
- If processing of the EID is scheme specific, should we add rules for when the scheme is unknown ? String based-longest-prefix matching ?

Implementation Decision: LinkLocal CLA-EID (1/2)

- A CLA-Specific EIDs is an unambiguous 1-hop **Endpoint Identifier** and an **Interface Identifier** that directly maps to a Convergence Layer Interface (or Channel). cla:<cla-scheme>:<cla-specific-part>
- Example :
  - cla:stcp:192.168.1.3:4559
  - cla:tcpclv4:10.127.35.2:4558
  - cla:usb:1bcf:2b91:bundlefolder
  - cla:mail:lucien@rightmesh:[dtn-bundle]
- CLA-EIDs are unambiguous and can also be parsed to instantiates a Channel proactively.

Implementation Decision: LinkLocal CLA-EID (2/2)

- Since it is also an EID, it can be used as a source or a destination in Bundle for 1-hop transmission.
- **DRAFT modification proposal in 7.2. (Summary of Convergence Layer Services) add the following service:**

  - if channel can send bundles, MUST provides a singleton CLA-EID identifying unambiguously the bundle node that is reachable via the convergence protocol.

  - if channel can receive bundle, MUST provide a singleton CLA-EID identifying the local bundle node.

→ The Link-Local Table is thus a CLA-EID list of all the open Convergence Layer Channel

Implementation Decision : Routing Table to Resolve EID

- Contains static mappings (EID1 → EID2), that serves as a route locator provider
- Can also be fed by a « Passive Routing Module » (Prophet/Neighbor Discovery/etc.)
- **Resolution of an EID**: recursively follow the mapping to a set of CLA-EID
- Example :

dtn:node1	→	dtn:gateway
dtn:node3	→	cla:tcpclv4:10.11.12.13:4557
dtn:node1	→	dtn:node2
dtn:node2	→	dtn:node3
*	→	cla:stcp:1.2.3.4:4556

- Resolving « **dtn:node1** » would give :

```
ForwardSet = { cla:tcpclv4:10.11.12.13:4557, cla:stcp:1.2.3.4:4556 }
```

Implementation Decision: Routing Logic Priority (1/2)

1. Lookup the Local-EID Table for delivery (already in draft-ietf-bpbis section 5.2 – step1)
2. **DRAFT Modification Protocol in 5.4 Bundle Forwarding, add the following rules 2.1 and 2.2 :**
   1. Lookup the Link-Local Table for 1-hop direct forwarding.
   2. Lookup Routing Table (resolve to CLA-EIDs → ForwardSET)

^

2.1 and 2.2 is the default routing (include into bpbis?) / 2.3 and 2.4 is extension-based routing

V

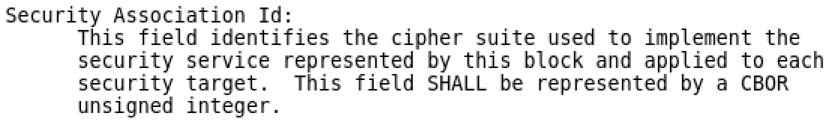
3. Lookup for a « routing module » (if any) – for example, if the bundle carries its own routing strategy inside a RoutingBlock (BIB/GeographicalRouting/others) then transfer the custody of the bundle to the relevant module.
4. Otherwise goes into cold Storage (bundle might be deleted if unavailable)

-> `Watch the bundle for relevant events`

-> Triggers the ConnectionAgent to try and force opportunities with the ForwardSET (todo: define policies to prevent abuse)

Hackathon – BPSec impl report

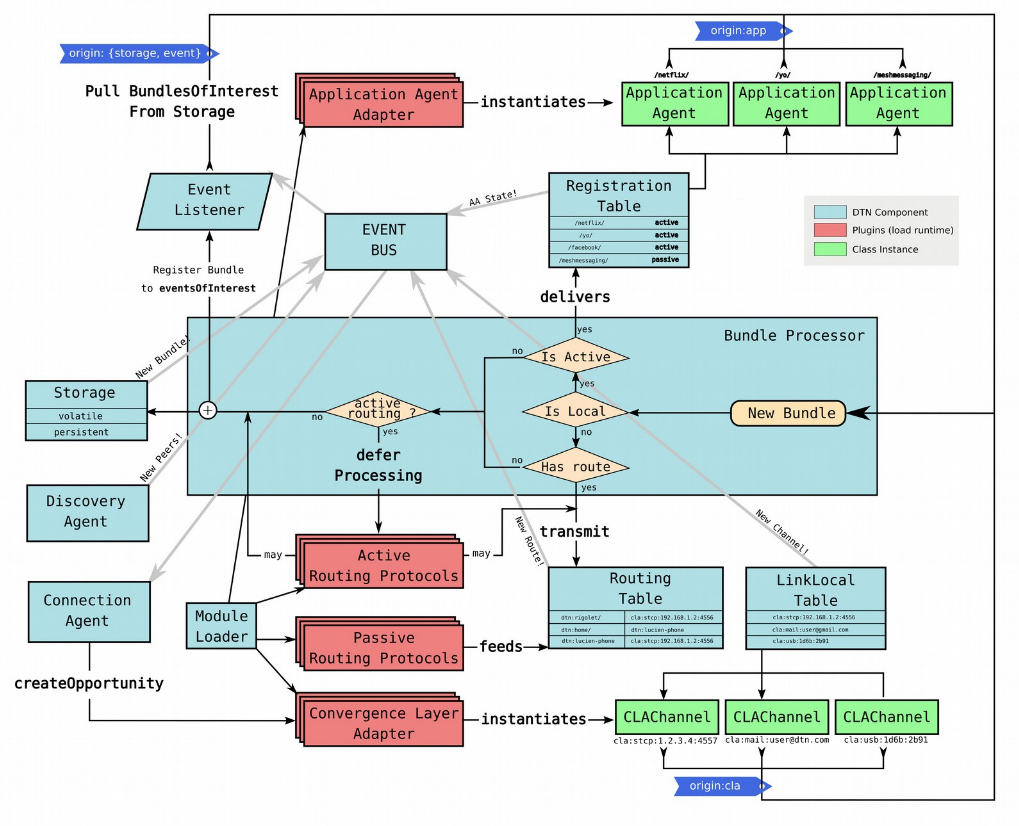
- Implementation :
  - draft-ietf-dtn-bpsec-08
  - draft-birrane-dtn-bpsec-interop-cs-03
- Overall a clean draft, relatively easy to implement !
- Proposition :
  - Add a **BLOCK_IS_ENCRYPTED** or **BLOCK_IS_UNINTELLIGIBLE** into BlockHeader in draft-ietf-dtn-bpbis (to avoid trying to parse an encrypted block payload)

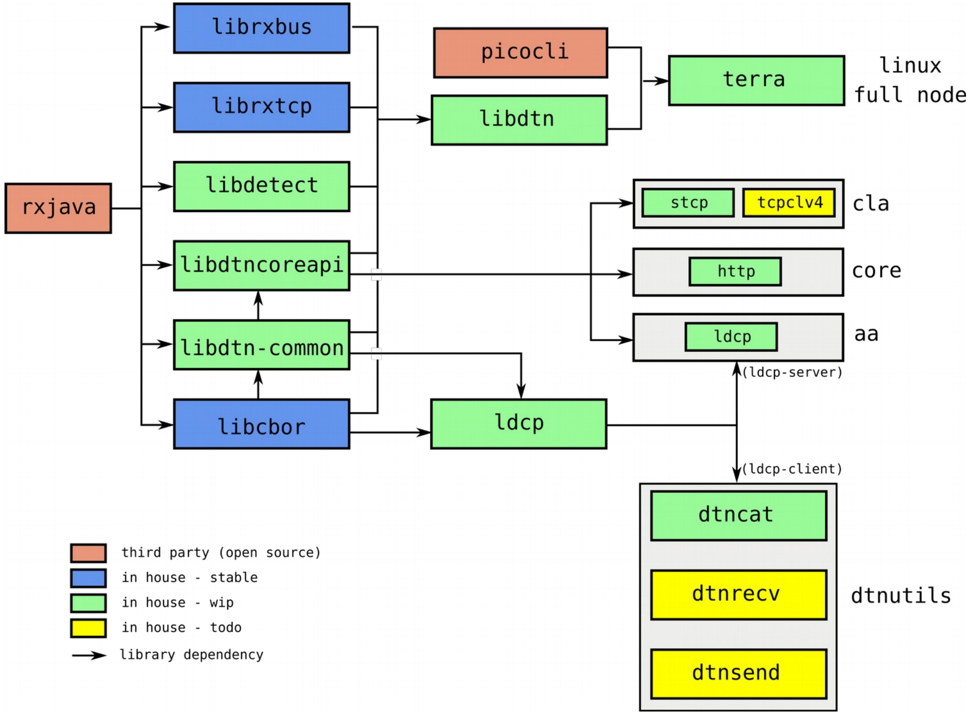


```
Security Association Id:
      This field identifies the cipher suite used to implement the
      security service represented by this block and applied to each
      security target.  This field SHALL be represented by a CBOR
      unsigned integer.
```

  - In **Abstract Security Block** : Replace Security Association Id with Cipher Suite Id, it is confusing with the Security Association Id from Security Association Parameter, or does it represent the parameters security association id ?
  - In **Abstract Security Block** : though implicit, it isn’t stated that it should be a CBOR array.
  - In **SecurityResults**, each individual results SHOULD be a TLV rather than a TV (to prevent running a generic parser if cipher is unknown)

Current Software Architecture





November 8, 2018

Terra - RESTful core AAs

- Passive Routing Module : 1-hop contact information
  - > **registers to** api:me/hello/
  - > **sends to** <new-peer-cla-eid>/hello/

when a new Convergence Layer Channel is open, Terra sends a « HELLO » Bundle that contains the list of all EIDs it is registered to.

Upon reception of such HELLO bundle, the hello service populate the Routing Table and maps all received EID with the CLA-EID from the bundle was received.

Thanks for your attention

- Questions ?